**AI학습용데이터활용 프로젝트 설명서**

20185231 김지환

1. 주제

|  |  |
| --- | --- |
| **주제** | 일몰 시간 데이터 수집 및 분석 |
| **선정**  **이유** | 최근 모바일 기기나 어플리케이션에서 자동적으로 일몰시간에 맞춰서 작동되는 ‘다크모드’와 일출시간부터 일몰시간까지 ‘라이트모드’로 구성이 되어있기에 일몰할 때 스스로 어떻게 ‘다크모드’로 변환이 되는지 궁금증이 생겨 ‘다크모드’의 실행 시간, 즉, 일몰시간 데이터를 수집을 하고 싶어서 선정하게 되었다. |
| **개요** | ‘공공데이터포털’에서 ‘한국천문연구원\_출몰시각 정보’의 오픈API를 이용하여 정적 크롤링를 진행한다. 데이터를 지역과 해당 년도를 입력하여, 매월 말일의 일몰시간으로 가져와서 json과 csv로 저장한다. 크롤링한 데이터를 이용하여 각 매달 말일을 x축, 해당 일몰시간을 y축으로 지정하여 어느 때가 가장 일몰시간이 늦는 지 확인한다. |
| **분석 방법** | Regression(예측) |

1. 데이터

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **데이터 제공** | [한국천문연구원\_출몰시각 정보 | 공공데이터포털 (data.go.kr)](https://www.data.go.kr/tcs/dss/selectApiDataDetailView.do?publicDataPk=15012688) | | | | |
| **수집 방법** | 정적 크롤링 | | 데이터 포맷 | XML | |
| **요청 url** | http://apis.data.go.kr/B090041/openapi/service/RiseSetInfoService/getAreaRiseSetInfo | | | | |
| **데이터 설명** | 날짜와 지역을 기준으로 일출, 일남중, 일몰, 월출, 월남중, 월몰, 시민박명(아침), 시민박명(저녁), 항해박명(아침), 항해박명(저녁), 천문박명(아침), 천문박명(저녁) 등의 정보를 제공한다.  2016년 기준으로 저장되어 있는 지역은 강릉, 강화도, 거제, 거창, 경산, 경주, 고성(강원), 고양, 고흥, 광양, 광주, 광주(경기), 구미, 군산, 김천, 김해, 남원, 남해, 대관령, 대구, 대덕, 대전, 독도, 동해, 마산, 목포, 무안, 밀양, 변산, 보령, 보성, 보현산, 부산, 부안, 부천, 사천, 삼척, 상주, 서귀포, 서산, 서울, 서천, 성산일출봉, 세종, 소백산, 속초, 수원, 순천, 승주, 시흥, 아산, 안동, 안산, 안양, 양양, 양평, 여수, 여수공항, 여주, 영광, 영덕, 영월, 영주, 영천, 완도, 용인, 울릉도, 울산, 울진, 원주, 의성, 익산, 인천, 임실, 장수, 장흥, 전주, 정읍, 제주, 제주(레), 제천, 주문진, 진도, 진주, 진해, 창원, 천안, 청주, 청주공항, 추풍령, 춘양, 춘천, 충주, 태백, 태안, 통영, 파주, 평택, 포항, 해남, 화성, 흑산도 이다. 매년 계산하는 지역은 크게 바뀌지 않는다. | | | | |
| **요청 메시지 명세** | | | | | |
| **항목명(영문)** | | **항목명(국문)** | | | **샘플데이터** |
| locdate | | 날짜 | | | 20200101 |
| location | | 지역 | | | 춘천 |
| **응답 메시지 명세** | | | | | |
| **항목명(영문)** | | **항목명(국문)** | | | **샘플데이터** |
| locdate | | 날짜 | | | 20200101 |
| location | | 지역 | | | 춘천 |
| longitude | | 경도 | | | 3700 |
| longitudeNum | | 경도(실수) | | | 126.9833330 |
| latitude | | 위도 | | | 12800 |
| latitudeNum | | 위도(실수) | | | 37.5666660 |
| sunrise | | 일출 | | | 074648 |
| suntransit | | 일중 | | | 123519 |
| sunset | | 일몰 | | | 172357 |
| moonrise | | 월출 | | | 142856 |
| moontransit | | 월중 | | | 213246 |
| moonset | | 월몰 | | | 034057 |
| civilm | | 시민박명(아침) | | | 071736 |
| civile | | 시민박명(저녁) | | | 175309 |
| nautm | | 항해박명(아침) | | | 064453 |
| naute | | 항해박명(저녁) | | | 182553 |
| astm | | 천문박명(아침) | | | 061304 |
| aste | | 천문박명(저녁) | | | 185742 |
| **요청 URL** | | | | | |
| <http://apis.data.go.kr/B090041/openapi/service/RiseSetInfoService/getAreaRiseSetInfo?location=%EC%84%9C%EC%9A%B8&locdate=20150101&ServiceKey>=인증키 | | | | | |

1. 코드해석

|  |
| --- |
|  |
| * 크롤링을 위한 라이브러리 import * import xmltodict: 이 데이터는 xml로 구성되어 있기에 xml을 dictionary 자료구조로 변형시켜줄 라이브러리 |
|  |
| * 매개변수로 받은 url을 받아서 urllib로 요청한다. * 요청과 함께 존재하면 현재시간과 같이 출력한다. |
|  |
| * 요청 URL 양식에 맞게 기본 URL에 입력할 위치(매개변수: location)와 날짜(매개변수: locdate), 받아온 인증키를 순서대로 넣어준다. 특히 위치(location)를 url에 기입할 때 한글로 입력하기 때문에 주소에 맞게 들어갈 아스키코드 형식으로 인코딩해준다. * 완성된 url을 getRequestUrl()에 넣어주고 반환된 값이 존재한다면 xml형식으로 반환이 됐을 것이다. 그렇기에 xmltodict.parse( )을 사용하여 xml형식을 dictionary로 변환해주고 json.dumps( )로 json형식 파일로 변환해준다. |
|  |
| * 매개변수 city와 date로 json파일에서 알맞은 데이터를 찾는다. * for문을 이용하여 1월부터 12월까지 구하고 알맞은 말일을 day에 기입하여 for문을 이용하여 해당 년도 365일 전부의 일몰시간을 구해준다. 그리고 위치(city)와 날짜(yyyymm)를 가지고 json파일에서 [‘response’][‘body’][‘items’][‘item’]을 거쳐서 원하는 데이터, 일몰시간을 구해준다. ex) [‘reponse’][‘body’][‘items’][‘item’][‘sunset’] (일몰시간 데이터) * 뽑아온 sunset데이터를 time에 맞춰 포맷 후 변수에 저장한다. * 뽑은 데이터들을 jsonResult에 dictionary형식으로 넣어주고 result list에 넣어준다. |
|  |
| * city와 date변수에 검색하고 싶은 값을 넣어 getService함수에 변수로 넣어준다. 그리고 반환된 jsonResult, result로 json파일과 csv파일로 만들어준다. |
|  |
| * main함수를 실행시켜주고 반환될 값들을 받아준다. |
|  |
| * 차트를 그리기 위해 matplotlib.pyplot을 import해주고 한글 사용을 위해 plt.rc를 사용한다. |
|  |
| * 데이터프레임에 있는 ‘일몰시간’은 문자열(datetime)로 저장되어 있기 때문에 ‘일몰시간\_int’라는 열을 새로 만들어서 날짜와 시간을 합쳐서 time형으로 포맷하여 시간형 데이터를 숫자형 데이터로 변환하여 넣어준다. |
|  |
|  |
| * Train 데이터와 test 데이터를 7:3으로 나눠서 사이킷런을 이용하여 regression진행 |
|  |
|  |